

OS8 Swift Cheat Sheet

iOS8 Swift Cheat Sheet and Quick Reference Guide for iPhone Developers. Swift is the new programming language used in developing applications for Mac OS and iOS, introduced by Apple in 2014. Swift is not, at present a replacement for Objective-C. This Cheat Sheet was Submitted by Yaffari over at iOS-Blog and has credits to: <http://kpbp.github.io/swiftcheatsheet/>

Resources & Tutorials

- [Swift Tutorials](#)
- [Official Swift iBook](#)
- [Swift Google+ Community](#)
- [Swift Development Course](#) - Coming soon, Sign up for updates

Arrays

```
var colors = ["red", "blue"]
var moreColors: String[] = ["orange", "purple"] // explicit type
colors.append("green") // [red, blue, green]
colors += "yellow" // [red, blue, green, yellow]
colors += moreColors // [red, blue, green, yellow, orange, purple]
```

```
var days = ["mon", "thu"]
var firstDay = days[0] // mon
days.insert("tue", atIndex: 1) // [mon, tue, thu]
days[2] = "wed" // [mon, tue, wed]
days.removeAtIndex(0) // [tue, wed]
```

Classes

```
class Counter {  
    var count: Int = 0  
    func inc() {  
        count++  
    }  
    func add(n: Int) {  
        count += n  
    }  
    func printCount() {  
        println("Count: \(count)")  
    }  
}  
  
var myCount = Counter()  
myCount.inc()  
myCount.add(2)  
myCount.printCount() // Count: 3
```

Conditionals

```
//IF STATEMENT  
let happy = true  
if happy {  
    println("We're Happy!")  
} else {  
    println("We're Sad :(")  
}  
// We're Happy!
```

```
let speed = 28  
if speed <= 0 {  
    println("Stationary")  
} else if speed <= 30 {  
    println("Safe speed")
```

```

} else {
    println("Too fast!")
}
// Safe speed

//SWITCH STATEMENT
let n = 2
switch n {
    case 1:
        println("It's 1!")
    case 2...4:
        println("It's between 2 and 4!")
    case 5, 6:
        println("It's 5 or 6")
    default:
        println("Its another number!")
}
// It's between 2 and 4!

```

Constants

```

let myInt = 1
myInt = 2 // compile-time error!

```

Dictionaries

```

var days = ["mon": "monday", "tue": "tuesday"]
days["tue"] = "tuesday" // change the value for key "tue"
days["wed"] = "wednesday" // add a new key/value pair

```

```

var moreDays: Dictionary = ["thu": "thursday", "fri": "friday"]
moreDays["thu"] = nil // remove thu from the dictionary
moreDays.removeValueForKey("fri") // remove fri from the dictionary

```

Enums

```

enum CollisionType: Int {
    case Player = 1
    case Enemy = 2
}

```

```
var type = CollisionType.Player
```

For Loops

```
for var index = 1; index < 3; ++index {  
    // loops with index taking values 1,2  
}  
for index in 1..3 {  
    // loops with index taking values 1,2  
}  
for index in 1...3 {  
    // loops with index taking values 1,2,3  
}
```

```
let colors = ["red", "blue", "yellow"]  
for color in colors {  
    println("Color: \$(color)")  
}  
// Color: red  
// Color: blue  
// Color: yellow
```

```
let days = ["mon": "monday", "tue": "tuesday"]  
for (shortDay, longDay) in days {  
    println("\$(shortDay) is short for \$(longDay)")  
}  
// mon is short for monday  
// tue is short for tuesday
```

Functions

```
func iAdd(a: Int, b: Int) -> Int {  
    return a + b  
}  
iAdd(2, 3) // returns 5
```

```
func eitherSide(n: Int) -> (nMinusOne: Int, nPlusOne: Int) {  
    return (n-1, n+1)  
}  
eitherSide(5) // returns the tuple (4,6)
```

Logical Operators

```
var happy = true
var sad = !happy // logical NOT, sad = false
var everyoneHappy = happy && sad // logical AND, everyoneHappy = false
var someoneHappy = happy || sad // logical OR, someoneHappy = true
```

Printing

```
let name = "swift"
println("Hello")
println("My name is \(name)")
print("See you ")
print("later")
/*
    Hello
    My name is swift
    See you later
*/
```

Strings

```
var myString = "a"
let myImmutableString = "c"
myString += "b" // ab
myString = myString + myImmutableString // abc
myImmutableString += "d" // compile-time error!
```

```
let count = 7
let message = "There are \(count) days in a week"
```

Variables

```
var myInt = 1
var myExplicitInt: Int = 1 // explicit type
var x = 1, y = 2, z = 3 // declare multiple integers
myExplicitInt = 2 // set to another integer value
```